## Design of High Speed 16x16 Bit MAC Units using Vedic Multiplier

Vikas Gupta, PhD HOD and Professor Department of ECE Technocrats Institute of Technology, Bhopal

#### ABSTRACT

The MAC architecture of Vedic multiplier with 'Urdhavatiryakbhyam' methodology for 16 bit MAC using Vedic multiplier is proposed. Equations for each bit of 32 bit resultant are calculated distinctly. They are chosen as they decrease vertical critical delay in comparison to the conventional architectures of MAC implemented using half adders only and so make the multiplier fast. The designs are coded in Verilog HDL and synthesized with Xilinx ISE 14.6 using virtex series of FPGA (Field Programmable Gate Array). The combinational delay calculated for proposed  $16 \times$ 16 bit multiplier is 10.50 ns. Further speed comparisons of compressor adders with traditional ones and proposed multiplier with popular methods for multiplication are shown. Results clearly indicate the better speed performance of our proposed Vedic multiplier

#### **Keywords**

Multiply and Accumulate, Vedic Multiplier, Verilog HDL, half adder.

## **1. INTRODUCTION**

Digital multipliers are the main components of all Digital signal processors. The speed of DSP is mainly considered by the speed of its multipliers architecture. Multiply Accumulate (MAC) operation is a commonly used operation in various Digital Signal Processing Applications. Use of a Digital Signal processor can significantly increase the performance of a MAC. Normally a multiply accumulate unit consists of a multiplier along with an accumulator which stores previous multiplication products. Since system performance widely depends on time needed to execute the instruction and multiplication being the most time consuming, any improvement to multiplication will inherently improve the system performance. Multiplication can be designed using several algorithms such as compressor adder, ALU, Modified Booth algorithm and Wallace tree. In array multiplier multiplication of two numbers can be obtained with one micro operation. It is a fast method of multiplication since the only delay is time for the signals to propagate through the gates. But it requires larger number of gates and so it is less economical. A new algorithm is developed that uses Vedic mathematics. The conventional mathematical algorithms can be simplified and even optimized by the use of Vedic mathematics. The Vedic algorithm is applicable to arithmetic, trigonometric, plain and spherical geometry, calculus. The whole of Vedic mathematics is based on 16 sutras. Here we use Urdhva Tiryagbhyam of Vedic mathematics. This sutra was traditionally used in ancient for the multiplication of two decimal numbers in relatively less time [1]. The architecture of urdhva tiryagbhyam is explained that any N x N multiplication can be efficiently designed by breaking it into smaller numbers of size (N/2=n) and these smaller numbers

Mukesh Kumar M.Tech Student Department of ECE Technocrats Institute of Technology, Bhopal

can again break into smaller numbers (n/2) till we reach the multiplicand size of (2 x2). This work presents a systematic design methodology for fast and area efficient digital multiplier based on Vedic mathematics and then a MAC unit has been made which uses this multiplier.

## 2. MAC OPERATION

Multiply-accumulate operation is one of the basic arithmetic operations extensively used in modern digital signal processing (DSP). Most arithmetic, such as digital filtering, convolution and fast Fourier transform (FFT), requires highperformance multiply accumulate operations. The multiplyaccumulator (MAC) unit always lies in the critical path that determines the speed of the overall hardware systems. Therefore, a high-speed MAC that is capable of supporting multiple precisions and parallel operations is highly desirable.

Basically a MAC unit destines a fast multiplier fitted in the data path and the multiplied output of multiplier is fed into a fast adder which is set to zero initially. The result of addition is stored in an accumulator register. The MAC unit should be able to produce output in one clock cycle and the new result of addition is added to the previous one and stored in the accumulator register, Figure1.Below shows basic MAC architecture. Here the multiplier that has been used is a Vedic Multiplier built using Urdhva Tiryakbhyam Sutra and has been fitted into the MAC design. The MAC operation is the key operation, not only in DSP applications but also in several multimedia information processing and various other applications. As mentioned above, MAC unit consist of multiplier, adder and accumulator. The MAC inputs are obtained from the memory location and given to the multiplier block. The function of the MAC unit is given by the following equation: - [2]

F = sum (PiQi)....(1)

The general MAC architecture consists of a conventional multiplier, adder and an accumulator. Where the output is added to the previous MAC output result by an accumulate adder. The MAC unit is extensively used in FPGA, microprocessors and digital signal processors for dataintensive applications, such as filtering, convolution, and inner products of output. Most digital signal processing methods use nonlinear functions such as discrete cosine transform (DCT) or discrete wavelet transform (DWT) or FFT/IFFT computations that can be efficiently accelerated by dedicated MAC units. The 16 bit MAC unit here uses a 16x16 Vedic multiplier in its data path. As the 16x16 Vedic multiplier is faster than other multipliers for example compressor adder, even the MAC unit made using Vedic multiplier is faster than the booth and ALU. In the MAC unit, the data inputs, A and B are 16 bits wide and they are stored

in two data registers, that is Data a\_reg and Data b\_reg, both being 16 bits wide, Then the inputs are fed into a Vedic multiplier, which stores the result "Mult" in a 32 bit wide register named, Multiply\_reg. The contents of Multiply\_reg are continuously fed in to a conventional adder and the result is stored in a 64 bit wide register "Dataout\_reg". [3]



#### Figure 1

#### 3. VEDIC MULTIPLIER FOR 2X2

In 2x2 bit multiplier, the multiplicand has 2 bits each and the result of multiplication is of 4 bits. So in input the range of inputs goes from (00) to (11) and output lies in the set of (0000, 0001, 0010, 0011, 0100, 0110, 1001). A simple design show in given in figure2. By using Urdhva Tiryakbhyam, the multiplication takes place as illustrated in given figure. Here multiplication 3 Page | 31 is vertical multiplication of LSB of both multiplication and addition of the partial products. [4]

s0 = a0b0 ......(2) c1s1 = a1b0 + a0b1.....(3) c2s2 = c1 + a1b1.....(4)

The final result will be c2s2s1s0. This multiplication method is applicable for all the cases. The hardware realization of 2x2multiplier blocks is illustrated in Fig no 2. For the sake of simplicity, the usage of clock and registers is not shown, but emphasis has been laid on understanding of the algorithm.



I





Figure 2

#### 4. VEDIC MULTIPLIER FOR 4X4

The 4x4 Multiplier is made by using four 2x2 multiplier blocks. Here, the multiplicands are of bit size (n=4) where as the result is of 8 bit size. The input is broken into smaller chunks of size of n/2 = 2, for both inputs, that is a and b. These newly formed chunks of 2 bits are given as input to 2x2 multiplier blocks and the result produced 4 bits, which are the output produced from 2x2 multiplier block are sent for addition to an addition tree, as shown in the Figure 3. [5]

M = a3 a2 a1 a0 (multiplicand)

N = b3 b2 b1 b0 (multiplier)

This four bit input is divided into two bit numbers:

 $M1 = a3 \ a2 \ A0 = a1 \ a0 \ \&$  $N1 = b3 \ b2 \ B0 = b1 \ b0$ 



Figure 3

## 5. VEDIC MULTIPLIER FOR 8X8

The 8x8 multiplier is made by using 4, 4x4 multiplier blocks. Here the multiplicands are of bit size (n=8) and the output is of bit size 16. The input is broken into small units of size n/2=4. The newly formed units of 4 bits are given as the input of 4x4 multiplier block, where again these units is divided into even smaller units of size n/4=2 and fed to 2x2 multiply block. The output obtained from output of 4x4 bit multiply block which is of 8 bits are sent for addition to an addition tree. The block diagram is shown in Figure 4. Let the two numbers A and B

*M*= *a*7 *a*6 *a*5 *a*4 *a*3 *a*2 *a*1 *a*0 *N*= *b*7 *b*6 *b*5 *b*4 *b*3 *b*2 *b*1 *b*0

The input is divided into small bits of four bits.





**Bits input to Addition Tree** 



Figure 4

## 6. VEDIC MULTIPLIER FOR 16X16

The 16x16 bit multiplier is made by using 4,8x8 multiplier blocks. Here the multiplicand are of size (n=16) and the result obtained is 32 bit size. The input is broken into small unit of size n/2=8. These newly formed units are given as input to 8x8 multiplier blocks. Again the new units are broken into even smaller units of size n/4=4 and fed to 4x4 multiply block. The newly formed 4x4 bit unit is again divided in half to get unit of size 2, which is fed to a 2x2x multiply block. The result produced from output of 8x8 bit multiply block which is of 16 bits are sent for addition to an addition tree. Let the two numbers are

$$\begin{split} M &= a15 \; a14 \; a13 \; a12 \; a11 \; a10 \; a9 \; a8 \; a7 \; a6 \; a5 \; a4 \; a3 \; a2 \; a1 \; a0 \\ N &= b15 \; b14 \; b13 \; b12 \; b11 \; b10 \; b9 \; b8 \; b7 \; b6 \; b5 \; b4 \; b3 \; b2 \; b1 \; b0 \end{split}$$

The input is divided into small units:-

M1 = a15 a14 a13 a12 a11 a10 a9 a8 A0= a7 a6 a5 a4 a3 a2 a1 a0 & N1= b15 b14 b13 b12 b11 b10 b9 b8 B0= b7 b6 b5 b4 b3 b2 b1 b0

Again it is divided into small unit of size 4:-

 $\begin{array}{l} M1a = a15 \; a14 \; a13 \; a12, \; M1b = a11 \; a10 \; a9 \; a8, \\ M0a = a7 \; a6 \; a5 \; a4, \; M0b = a3 \; a2 \; a1 \; a0 \\ N1a = b15 \; b14 \; b13 \; b12, \; N1b = b11 \; b10 \; b9 \; b8, \\ N0a = b7 \; b6 \; b5 \; b4, \; N0b = b3 \; b2 \; b1 \; b0 \\ \end{array}$ 



Figure 5

# 7. RTL REPRESENTATION OF MAC UNIT

The RTL (Register Transfer Level) schematic of the 16x16 bit Vedic multiplier and MAC design shown in Figure 06



Figure 6

#### 7.1 Technology schematic representation

Technology schematic representation of 16x16 bit MAC architecture. It shows the uses of component for this design. Shown in figure no 7.



Figure 7

### 8. SIMULATION

It is observed that for MAC 16x16 bit Vedic multiplier the delay obtained are 10.504 ns. Isim by Xilinx is used for simulation and synthesis of the Vedic multiplier is carried out using Xilinx ISE 14.6. Hence the MAC using multiplier architecture is found to be most efficient in terms of speed. Figure 8 displays the simulation result of MAC 16x16 Vedic multiplier.



Figure 8

## 9. COMPARISON

Table No 1

16x16 bit Vedic multiplier	Proposed multiplier	With compressor adder	Booth multiplier
Delays (ns)	10.50 ns	32ns	37.04ns

#### **10. RESULT AND FUTURE WORK**

The architectures of 16 bit Vedic multiplier and MAC unit are designed in Verilog HDL language. Logic synthesis and simulation is done using EDA (Electronic Design Automation) tool in Xilinx ISE 14.6 simulator. Device used is virtex6:XC6UCX75T-2-ff484.Comparison XILINX, of combinational delay (ns) for Vedic multiplier in below table 1. A combinational delay of 10.50 ns is seen for proposed 16x16 bit Vedic multiplier architecture which is small in comparison to the Compressor adder, Wallace Tree and Booth multiplier. The higher speed of the multiplier is come across by the use of MAC unit architectures and reduces sum operations. The simulation results for proposed 16 bit Vedic multiplier with 'M' and 'N' as 16 bit input bits and 's' as an 32 output.

Table no 2

Algorithm	Proposed 16 bit multiplier	
Delay (ns)	10.50 ns	
Number of slices	64 out of 93120 (0%)	
Number of 4 input LUTs	563 out of 46560 (1%)	
Number of bounded IOBs	98 out of 240 (40%)	

### **11. FUTURE WORK**

Vedic Mathematics, gives us a clue of symmetric computation. All these methods are very efficient as far as manual calculations are concerned. If all those methods effectively implement hardware, it will reduce the computational speed drastically. Therefore, it could be possible to implement a complete ALU using all these methods using Vedic mathematics methods. Vedic mathematics is long been known but has not been implemented in the DSP and ADSP processors employing large number of multiplications in calculating the various transforms like FFTs and the IFFTs. By using these ancient Indian Vedic mathematics methods world can achieve new heights of performance and quality for the cutting edge technology devices.

#### **12. REFERENCES**

- [1] A novel high-speed approach for  $16 \times 16$  Vedic multi plication with compressor adders doi:10.1016/j.compeleceng.2015.11.006.
- [2] Saokar SS, Banakar RM. High-speed signed multiplier for digital signal processing applications. In: Proceedings of signal processing, computing and control. doi:10.1109/ISPCC.2012.6224373.
- [3] Prakash AR, Kirubaveni S. Performance evaluation of FFT processor using conventional and Vedic algorithm. In: Proceedings of IEEE conference on emerging trends in computing, communication and nanotechnology (ICE-CCN); 2013. p. 89–94. doi:10.1109/ICE-CCN.2013.6528470.
- [4] A high-speed block convolution using ancient Indian Vedic mathematics. In: Proceedings of IEEE conference on computational intelligence and multimedia applications (ICCIMA); 2007. p. 169–73. doi:10.1109/ICCIMA.2007.332.
- [5] M. Morris Mano, "Computer System Architecture", 3<sup>rd</sup> edition, Prientice-Hall, New Jersey, USA, 1993, pp. 346-348.